

RECEIVED
CENTRAL FAX CENTER

NOV 01 2007

AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions, and listings, of claims in the application.

1-20 (Cancelled)

21. (Currently Amended) A method comprising:

determining whether an instruction for a first thread is of a first type;

pausing processing of instructions of the first thread upon determining that the first instruction is of the first type while processing instructions from a second thread; and

resuming processing of a second ~~microinstruction~~ instruction decoded from the instruction after execution of a first ~~microinstruction~~ instruction decoded from the instruction.

22. (Currently Amended) The method of claim 21 further comprising decoding the instruction into the first ~~microinstruction~~ instruction and the second ~~microinstruction~~ instruction.

23. (Currently Amended) The method of claim 22 wherein the first ~~microinstruction~~ instruction causes a value to be stored in memory for the first thread.

24. (Currently Amended) The method of claim 23 further comprising:

processing the second ~~microinstruction~~ instruction for execution when the value stored in the memory is reset.

25. (Currently Amended) The method of claim 24 wherein the value stored in the memory is reset when the first ~~microinstruction~~ instruction is retired.

26. (Previously Presented) A method comprising:

determining whether an instruction of a first thread is of a first type;

initiating a counter upon determining that the instruction is of the first type, wherein the instruction includes an operand and the initiating includes loading the counter with the operand; and

pausing processing of instructions of the first thread until the counter reaches the value while processing instructions of a second thread at the pipeline stage.

27. (Cancelled)

28. (Previously Presented) The method of claim 26 further comprising resuming processing instructions of the first thread after the counter reaches the value.

29. (Currently Amended) An apparatus, comprising:

a decode unit to determine whether an instruction of a first thread is of a first type, said decode unit to pause processing of instructions of said first thread upon determining that the instruction is of the first type by generating a first ~~microinstruction~~ instruction to cause a value to be stored in a memory for the first thread while instructions from a second thread can be processed, said decode unit further to generate a second ~~microinstruction~~ instruction upon which processing is to resume after execution of the first ~~microinstruction~~ instruction.

30. (Cancelled)

31. (Cancelled)

32. (Currently Amended) The apparatus of claim 29 wherein the decode unit is to process the second ~~microinstruction~~ instruction when the value stored in the memory is reset.

33. (Currently Amended) The apparatus of claim 32 further comprising:

a retire unit coupled to the decode unit wherein the retire unit is to cause the value stored in the memory to be reset when the first ~~microinstruction~~ instruction is retired by the retire unit.

34. (Previously Presented) An apparatus comprising:

a decode unit to determine whether a instruction for a first thread is of a first type;

a counter coupled to the decode unit, the counter to be loaded with a value of an operand of the instruction if the first instruction for the first thread is of the first type, the decode unit to pause processing instructions of the first thread until the counter reaches a the value; and

wherein instructions for a second thread can be processed while instructions of the first thread are paused from being processed.

35. (Cancelled)

36. (Previously Presented) The apparatus of claim 34 wherein the decode unit is to operate while the first thread is paused from being processed.

37. (New) The apparatus of claim 29, implemented in a system including operating system code to cause the instruction to be forward to the decode unit.

38. (New) The apparatus of claim 34, implemented in a system including operating system code to cause the instruction to be forward to the decode unit.

39. (New) An apparatus comprising:

a queue; and

a decode unit coupled with the queue to receive and decode a pause instruction of a first thread, the decode unit in response to the pause instruction to pause processing of instructions

of the first thread for a period of time while instructions from a second thread are processed and the decode unit in response to the pause instruction to generate an instruction that is to be queued in the queue during the period of time.

40. (New) An apparatus comprising:

a decode unit to receive and decode a pause instruction of a first thread, the decode unit in response to the pause instruction to pause processing of instructions of the first thread while instructions from a second thread are processed by generating a first instruction and a second instruction, the first instruction to set a flag and the second instruction to prevent the instructions of the first thread from being processed until after the first instruction resets the flag.

41. (New) A pause instruction stored in one of a memory and a queue that is operable to cause a processor to perform operations comprising:

decoding the pause instruction;

loading a value of an operand of the pause instruction into a counter; and

pausing processing of instructions of a first thread from which the pause instruction was received while instructions of a second thread are processed until the counter reaches the value.